AFRL-IF-RS-TR-2003-232
Final Technical Report
October 2003

MEASURING QUALITY OF INFORMATION ASSURANCE (QoIA)

Pennsylvania State University

Sponsored by
Defense Advanced Research Projects Agency
DARPA Order No. N899

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

The views and conclusions contained in this document are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the U.S. Government.

AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-2003-232 has been reviewed and is approved for publication.

APPROVED: /s/
JOHN C. FAUST
Project Engineer

FOR THE DIRECTOR: /s/
WARREN H. DEBANY, JR., Technical Advisor
Information Grid Division
Information Directorate

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 074-0188*

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503

1. AGENCY USE ONLY (Leave blank)	**2. REPORT DATE** OCTOBER 2003	**3. REPORT TYPE AND DATES COVERED** Final Aug 02 – Feb 03

4. TITLE AND SUBTITLE
MEASURING QUALITY OF INFORMATION ASSURANCE (QoIA)

5. FUNDING NUMBERS
C - F30602-02-1-0216
PE - 61101E
PR - N899
TA - A1
WU - 06

6. AUTHOR(S)
Peng Liu

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
The Pennsylvania State University
School of Information Sciences and Technology
University Park Pennsylvania 16801

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Defense Advanced Research Projects Agency	AFRL/IFGB
3701 North Fairfax Drive	525 Brooks Road
Arlington Virginia 22203-1714	Rome New York 13441-4505

10. SPONSORING / MONITORING AGENCY REPORT NUMBER
AFRL-IF-RS-TR-2003-232

11. SUPPLEMENTARY NOTES

AFRL Project Engineer: John C. Faust/IFGB/(315) 330-4544/ John.Faust@rl.af.mil

12a. DISTRIBUTION / AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 Words)*
Current information assurance techniques do not allow us to state quantitatively how assured our systems and networks are. As a result, (a) security and assurance measures can only be designed and built into information systems in an ad hoc fashion, (b) it is difficult to characterize the capabilities of security measures, and (c) information systems cannot deliver quality of information assurance (QoIA) guarantees. This seedling project had two objectives: (1) to explore an economics theoretic framework for measuring assurance and (2) to explore a theory of QoIA management. For each objective, the study defines the problem space, offers some potentially feasible solutions, and creates a technology development roadmap for a 5 to 7 year time horizon. The key idea is to use incentive-based, economic models of attacker intent, objectives and strategies (AIOS) to measure a system's overall assurance capacity. As a result, a preliminary framework for AIOS modeling and inference is developed along with an approach which uses AIOS inferences to measure a system's assurance capacity. Two real-world assurance measuring case studies were conducted. Finally, a preliminary framework for measuring QoIA and delivering QoIA services in mission critical database systems is proposed.

14. SUBJECT TERMS
Information Assurance Measurement, Quality of Information Assurance, Economic Models of Attacker Intent, Game-Theoretic Models for Assurance, Nash Equilibrium Strategies

15. NUMBER OF PAGES
36

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	**18. SECURITY CLASSIFICATION OF THIS PAGE**	**19. SECURITY CLASSIFICATION OF ABSTRACT**	**20. LIMITATION OF ABSTRACT**
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

Table of Contents

Contents

List of Figures

1 Introduction

Current information assurance techniques do not allow us to state quantitatively how assured our systems and networks are. Without quantitative statements about assurance: (a) people cannot have a tangible understanding about how assured their systems and networks are, (b) it is difficult to characterize the capabilities of protective, detection, reactive, proactive or self-regenerative security measures, such as firewalls, intrusion detection systems, self-healing techniques, design diverse redundancy, proactive secret sharing, and deception, (c) it is difficult for people to compare the capabilities of two security measures; it is difficult for people to compare the assurance of two secure information systems, (d) people cannot find a tangible correlation between a qualitative security evaluation statement and the amount of assurance they actually get, (e) security and assurance measures can only be designed in an ad hoc fashion, based solely on what feels right, as opposed to whether the design can meet a quantitatively stated assurance requirement, (f) security and assurance can only be built into information systems in an ad hoc fashion, based solely on what can be afforded or what feels right, as opposed to what is desired or required for a given application and its operating environment. There is no guarantee that systems designed as such will be effectively protected when under a sustained cyber attack.

The key to solve the above problems is the idea of “Measuring Assurance in Cyber Space”, where measures of merit and metrics to characterize quantitatively various dimensions of security (availability, integrity, confidentiality, authentication, and non-repudiation) are identified, modeled, measured, monitored, evaluated, and controlled. If this idea bears fruit, researchers and designers of information system security will be able to make quantitative evaluations of novel architectural approaches, perform cost-benefit trade-offs, and create designs that meet specified levels of assurance.

We believe the impact of the research on measuring information assurance will go beyond measuring. First, to quantitatively measure assurance, we need to quantitatively model secure information systems and the relevant IA domain issues. Hence the research on measuring IA can motivate new cyber security models. Second, to measure an information system’s resilience against intentional, well-planned attackers who issue non-random attacks, we need to model attacker intent, objectives, and strategies. Hence the research on measuring IA can motivate new attacker models. Third, since quantitative assurance measurements can give people great leverage in designing better secure systems, the research on measuring IA can revolutionize security design methodologies. Finally, the research on measuring IA can revolutionize the way people evaluate security and assurance, the way vendors promote their products, and the way people deploy security and assurance measures.

1.1 Background and Prior Work

Previously, NSA's Orange Book and others in the rainbow series of books were the primary guidance documents for security design and evaluation. The current version is something called Common Criteria. However, these security-measuring techniques take a qualitative approach, and as a result current information assurance techniques do not allow us to state quantitatively how assured our systems and networks are. We cannot quantify the ability of protective security measures such as firewalls, virtual private networks, and boundary controllers to keep intruders out. It is difficult to characterize the capabilities of Intrusion Detection Systems to detect novel attacks. And the benefits of novel response mechanisms being researched for intrusion tolerance such as design diverse redundancy, proactive secret sharing, and deception techniques such as fingerprint masking, cannot be measured comparatively or on an absolute scale.

1.2 Objective

This seedling project has two objectives.

1. To explore an economics theoretic framework for measuring assurance. This framework will not only give us a quantitative approach to infer an information system's assurance capacity, but also gives us (a) an expressive language to express IA domain issues, (b) a novel family of cyber security models, (c) an incentive-based model of attack intent, objectives, and strategies, (d) a novel approach to predict attack actions, (e) a new methodology of *active defense*, (f) a new methodology in security design, and (g) a new method in security evaluation.

2. To explore a theory on QoIA management. Existing information assurance measuring is basically *state-oriented*, that is, existing assurance measuring techniques focus on the overall healthiness of the state of an information system, which is the composition of the

state of each component or resource of the system (which is usually shared by many customers or users); however, what customers really care about is the amount of assurance delivered to their tasks or *services*, which may involve only a couple of system components. The goal of this theory is to fill this gap. This theory will achieve a transition from state-oriented information assurance measuring to *service-oriented* information assurance measuring. Moreover, the theory will also address how to satisfy quantitatively specified service assurance requirements in the face of sustained attacks.

1.3 An Economics Theoretic Framework for Measuring Assurance (ETMA)

The key idea of the ETMA framework is using incentive-based, economic models of attacker intent, objectives, and strategies (AIOS) to measure a system's (overall) assurance capacity. Compared with reliability measuring, a unique challenge to assurance measuring is that attacks are not *random*. As a result, the combination of a well-defined system, a complete threat, vulnerability, attack, and risk (TVAR) taxonomy (in terms of the system), a representative set of assurance metrics, a representative workload, and (even) a rich attack history on the system may not be enough to yield accurate measurements about the system's (overall) assurance capacity, since statistics about old attacks may not capture the characteristics of new attacks which are intentional and not random.

Measuring a system's assurance capacity needs the ability to measure the attacker's attack capacity, which is however inter-dependent on the system's defense capacity. Measuring the attacker's attack capacity needs the ability to model the attacker's IOS, which is however inter-dependent on the system's IOS.

An incentive-based, economic model of AIOS not only models the attacker's IOS but also models the system's IOS. Moreover, this new model uses game theory to model mathematically the inter-dependency between these two IOS. And such a mathematical model can generate valuable, quantitative inferences about both the attacker's attack capacity and the system's defense capacity. These inferences can then be used to generate valuable quantitative measurements about the system's assurance capacity. Note that this model seamlessly integrates system specifications, TVAR, assurance metrics, workload, and attacks.

In particular, the economics theoretic assurance measuring framework is composed of the following components. Note that they are related to each other. The modeling language is used to specify the family of cyber security models. The AIOS model is built on top of the family of cyber security models. The AIOS model uses IA metrics and the corresponding measurements to define the utility earned by either the attacker or the system.The assurance measurements generated by our framework are based on the minimum utilities that could be earned by the system when a specific set of equilibrium defense strategies are taken by the system. The set of equilibrium defense strategies are determined based on the AIOS model. AIOS inferences can be valuable attack action predictions. The assurance measurements generated by our framework provide a lot of useful hints for optimal security design, and such measurements are natural means of security evaluation.

- An expressive modeling language that can express a variety of IA domain issues.
- A family of novel cyber security models that model cyber security not only from the defense perspective, but also from the attack (or offense) perspective. To our best knowledge, this family is the first security model that can model how the attacker and the defender (i.e., the computer system) can interact dynamically with each other.
- An incentive-based, economic model of AIOS. This model can compute valuable AIOS inferences.
- An incentive-based, economic interpretation of IA metrics and measurements that are security mechanism independent.
- An economic, game-theoretic approach of assurance measuring, namely, using economic utilities and equilibrium AIOS inferences to measure a system's overall assurance capacity.
- An economic, game-theoretic approach to predict attack actions with confidence.
- An economic, game-theoretic method of optimal security design. This method can identify the key design issues and factors, and can evaluate the benefits of a new security design either comparatively or on an absolute scale.
- An economic, game-theoretic method of security evaluation. While existing security evaluation techniques are primarily qualitative, this method is quantitative.

A more tangible discussion of each of the components of our framework is presented in Section 2. So far, we have achieved the following:

- We have developed a preliminary framework for incentive-based, economic AIOS modeling. This framework includes an incentive-based attacker intent model, an incentive-based attacker objective model, and an incentive-based attacker strategies model. This framework includes a general game theoretic formalization which integrates the three models. This framework develops a tentative taxonomy of game theoretic formalizations for AIOS modeling. This taxonomy identifies the conditions under which a specific type of game theoretic formalization is the most effective. This framework also identifies the key challenges of economic assurance capacity inferring, namely incomplete information, uncertainty, and complexity, and their impact on the accuracy and cost-effectiveness of economic assurance capacity inferring.

- We have developed a preliminary framework for using incentive-based, economic AIOS inferences to measure the assurance capacity of a system.

- We have finished Phase I of an assurance capacity measuring case study regarding Internet DDoS attacks. The goal of this case study is to measure the assurance capacity of pushback-based Internet defense in terms of the *minimum bandwidth* of the good traffic that can reach the target. We use *ns2* to simulate both UDP and ICMP based DDoS attacks on a set of routers armed with the pushback module. We developed a specific Bayesian game formalization to model the battles between the DDoS attacker and the set of routers. Using the (simulation) measurements measured based on a set of settings of the routers and the (bad/poor/good) traffic, Phase I yields interesting assurance capacity measures when the defense strategy is rather fixed. In Phase II, we plan to incorporate a large space of defense strategies.

- We have done an assurance capacity measuring case study regarding credit card frauds. The goal of this case study is to measure the assurance capacity of credit card authorization systems armed with a fraud-detection sub-system. We developed a specific Bayesian game formalization to model the battles between frauds and a credit card company. This formalization develops a novel probabilistic uncertainty model between the fraud and the credit card company. Based on a simplified assumption about credit card frauds, the case study yields interesting measurements about the *maximum amount of money* that a customer could lose.

1.4 A Theory of QoIA Management

Existing assurance evaluation techniques seem to focus on system *states*. However, when a mission critical information system is deployed, what people really want are quantitative assurance guarantees in terms of a specific *service*. A service can be as simple as a database query, and as complex as a multi-phase task.

State assurance measures cannot be directly mapped to service assurance measures. The goal of service-oriented assurance measuring is to fill this gap. Service assurance measures provide a critical bridge from measuring system assurance to performing critical missions (and tasks) in an assured way.

In our theory, a service associated with a specific level of assurance requirements is called a *QoIA service*. The goal of the system is to ensure that the amount of assurance delivered together with a QoIA service will satisfy the corresponding assurance requirements. The life cycle of cost-effective delivery of QoIA services consists of at least 4 phases: (1) QoIA service reservation (via some service assurance measures); (2) mapping service assurance requirements to state assurance requirements; (3) QoIA service provision (through dynamic, differentiated, intelligent, state-assurance-based adaptations); (4) QoIA service validation. Note that every phase involves service or state assurance measuring.

It should be noticed that the QoIA management theory is built on top of assurance measuring and our ETMA framework. Without the ability to measure service assurance, QoIA services cannot be delivered. Without the ability to measure state assurance, service assurance cannot be measured. Without the ETMA framework, we cannot know the system's capacity in delivering QoIA services in the face of malicious attacks.

The QoIA management theory enhances the ETMA framework with the ability to deliver QoIA services. Although the ETMA framework is very powerful in measuring assurance capacity, the ETMA framework does not provide the ability to deliver QoIA services. Although the ability to measure assurance can dramatically improve people's understanding about systems' security and survivability, and the benefits of a specific security measure, we believe an ultimate goal of measuring assurance is to deliver QoIA services.

The QoIA management theory is composed of the following components.

- The concept of QoIA services and QoIA management.

- State assurance vs. service assurance; State assurance measuring vs. service assurance measuring; Static measuring vs. dynamic measuring.

- The QoIA service provision process: (1) QoIA service reservation (through IA requirements specifications); (2) Mapping service assurance requirements to state assurance requirements; (3) QoIA service provision through state-assurance-based QoIA adaptations; (4) QoIA service validation.

- Key techniques in delivering QoIA services: (a) Trustworthiness modeling and assessment; (b) Statistics-based state assurance measurement or estimation; (c) Intelligent QoIA adaptations; (d) Composite (and multi-level) QoIA adaptations; (e) Differential QoIA adaptations; differentiated trustworthiness maintenance; (f) Predictive QoIA adaptations.

- QoIA service validation: Measuring a system's capacity in delivering QoIA services.

- QoIA guided self-regeneration.

A more tangible discussion of each of the components of our QoIA management theory is presented in Section 3. So far, we have developed a preliminary framework for delivering QoIA services in mission critical database systems where a service is modeled as a database query. This framework covers the whole life cycle of QoIA-service delivery in database systems. This framework identifies a set of potentially feasible solutions to each phase of the life cycle, although the corresponding technical details are not completely worked out yet. For example, the framework proposes to specify (and measure) service assurance requirements through state assurance measures and the mapping that a query does from the database state to the set of results. The framework proposes a statistics-based approach to measure state assurance. The framework proposes to deliver service assurance through state assurance maintenance for cost-effectiveness. The framework proposes to deliver QoIA services through differentiated prevention, detection, and survivability controls. This framework proposes to deliver sustained

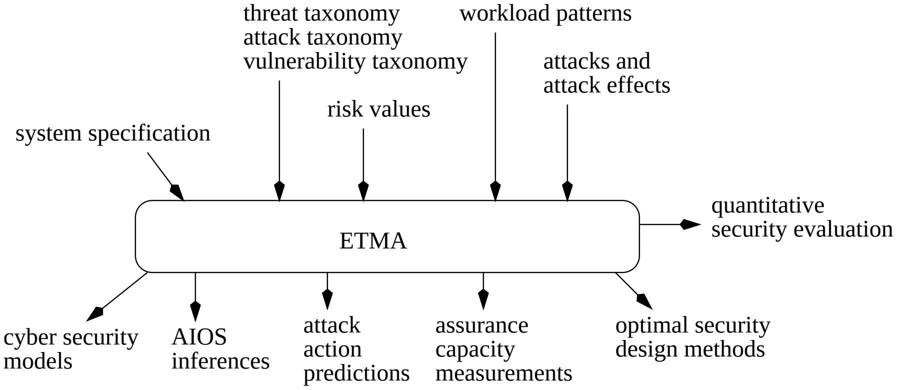


Figure 1: The Input-Output Semantics of the ETMA Framework

QoIA services through quantitative, predictive, intelligent, self-stabilizing, optimized, composite, state-assurance-based QoIA adaptations.

2 An Economic Framework for Measuring Assurance

2.1 Overview

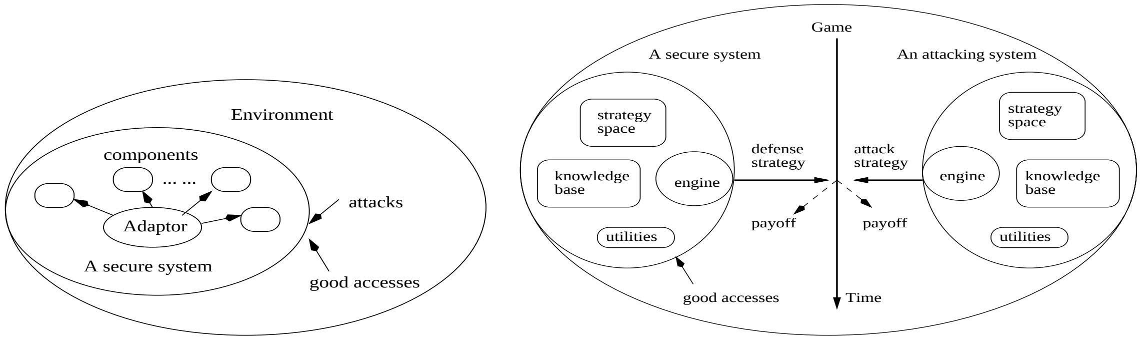
The input-output semantics of the ETMA framework is shown in Figure 1. The inputs of the ETMA framework cover every (important) element of information assurance, such as the system (including the set of protection measures), the attacker, the environment (e.g., the workload), vulnerabilities, threats, risks, attacks, attack effects, defense postures, and defense actions. On the other hand, the outputs of the ETMA framework not only give us quantitative assurance measurements, which are the primary goal of this seedling effort, but also give us a new security modeling methodology, a new security design methodology, a new attacker modeling methodology (i.e., AIOS modeling and inferring), a new attack prediction methodology, and a new security evaluation methodology at the same time.

2.2 How Can ETMA Model Cyber Security?

As shown in Figure 2(a), existing cyber security models focus on the system itself and consider attackers and attacks as a part of the *environment*, which also includes the legitimate accesses.

A serious drawback of this security design paradigm is that the attacker intent, objective, strategies are not taken into account. As a result, with little knowledge about AIOS, traditional cyber security models are unable to measure the system's resilience or assurance against intentional attacks that are not random. Moreover, systems designed based on such models can only *passively* detect and respond to the attacks, and passive security designs can seriously jeopardize the system's resilience.

A fundamental contribution of our framework is a new cyber security model shown in Figure 2(b), where the attacks are no longer a part of the environment, and AIOS modeling and security design are seamlessly integrated into one process. In particular, we model the attacker and the system as two *peer* systems, or two *players* fighting a series of *battles* or *game plays*, where (a) each player has a set of *strategies* to fight. A strategy can be an *action* or a sequence of actions. (b) The *strategy space* of the system is determined by the set of security facilities (or components) deployed to protect the system (Note that for clarity these components are not shown in Figure 2(b)). The system can defend against the attacker in many different manners by having multiple ways to configure its facilities. Each such manner can be a *defense strategy*. (c) The *strategy space* of the attacker is the set of attacks that the attacker is able to launch. An *attack strategy* can be an action or a sequence of actions. (d) At one point of time, the battle is defined by a pair of strategies: one from the attacker, one from the system. (e) The *outcome* of each battle indicates "who wins" in this round. In real world, an outcome could be "the attacker breaks in", "a malicious access request is rejected", etc. Note that for some battles, there may not be clear winners. (f) A battle-outcome yields two *utility* measures: one earned by the attacker, the other earned by the system. These utility measures indicate how the two players *prefer* the outcome. The framework uses utility measures to precisely define the meaning of "winning a game". (g) The goal of each player is to win the game, or to maximize his or her utilities. (h) The *environment* now only contains the good accesses. (i) Each player maintains a *knowledge base* to keep the player's knowledge about the other player and the other player's belief. (j) Each player selects the strategy to play based on his or her knowledge base. (k) Before each player fights a new battle, the outcomes of previous battles are already known, and become a part of each player's knowledge base. (l) The attacker's *uncertainty* about the system's defense, and the system's uncertainty about the attacker's offense, are all modeled by the rationality notion of an *expected-utility* maximizer. (m) The system's *uncertainty* about



(a) Traditional Cyber Security Models

(b) A New Cyber Security Model

Figure 2:

“whether or not the incoming access is an attack” is modeled by such techniques as having multiple *types* of players that play with the system.

A lot of technical details of our cyber security model can be found in [2], one of the four technical reports prepared for this seedling project.

This innovative attacker-system model and the results computed out of the model can enhance existing security design, evaluation, and deployment methodologies with three new abilities: (1) The ability to quantitatively model and infer AIOS. (2) The ability to quantitatively measure assurance. (3) The ability to do *active* defense, where the system no longer lags behind the attacker.

2.3 How Can ETMA Express IA Domain Issues?

Although it is widely recognized that given a system, many IA domain factors can contribute to assurance measuring, such as attackers, attacks, vulnerabilities, threats, risks, and security measures, existing assurance evaluations or measuring technologies cannot succinctly express the influence of each of the factors on assurance, or the relationships among these factors.

The cyber security model we proposed provides an expressive, economic language which can succinctly express both the influence of each IA factor on assurance (measuring) and the relationships among IA factors. For example, (a) the influence of an attack taxonomy on assurance measuring can be “expressed” as the influence of part of the attacker’s strategy space (i.e., the attack taxonomy) on the utility earned by the system. (b) The relationship between the attacker and the system can be “expressed” as a game played between the two parties. (c) The effects of a specific attack can be “expressed” as the difference between two specific utilities earned by the system or the attacker.

2.4 How Can ETMA Model and Infer Attacker Intent, Objective, and Strategies?

In [2], we present an economic, incentive-based approach to model and infer AIOS based on the cyber security model we just proposed. In particular,

- We use the concept of *incentives* to unify and quantify a variety of different *intents* of an attacker when he or she enforces an attack. Many kinds of intent can be modeled as incentives. For example, the amount of fun, the amount of profits, the amount of terror caused, the amount of political impact, the amount of satisfication when some challenges are taken, etc. Moreover, both the system’s and the attacker’s incentives can be measured by a set of IA metrics.

- We use the concept of *utilities* to integrate incentives and costs in such a way that attack *objectives* can be practically modeled. The utility earned by the attacker through an attack concerns both the incentives that the attacker gets and the costs that the attacker spends. Rational attackers want to maximize the incentives while minimizing the costs, given that a specific set of *constraints* are not violated. After the costs are quantified, utilities can be quantified through a distance function between the incentives and the costs.

- We use the concept of *strategies* to specify an intentional, well-planned attack, which is usually a sequence of attack actions. Strategies are taken to achieve objectives. Whether the attacker can achieve his or her objectives depends not only on his or her strategies but also on the system’s defense strategies.

- We use a novel game theoretic framework to integrate the IOS models of the attacker into an **AIOS inference machine**. This inference machine not only elegantly models the fundamental elements of AIOS, but also can produce valuable AIOS inferences. This inference machine models the interaction between the attacker and the system as one or more game plays where each player wants to maximize his or her total utilities, which represent his or her intent and objectives. Each game play involves two specific strategies from the attacker and the system, respectively. The outcome of each game play gives each player some utility. These game plays have a very important property, that is, they have Nash equilibrium strategies for both the attacker and the system. Nash equilibrium attack strategies are the "best" attack strategies for *rational* attackers because if the system always takes Nash equilibrium strategies, then the utilities earned by the attacker when he or she does not take any Nash equilibrium strategy will be smaller than the utilities earned by the attacker when he or she takes a Nash equilibrium strategy. For the same reason, Nash equilibrium strategies are also the best defense strategies for the system. Therefore, Nash equilibrium attack strategies and the corresponding utilities are realistic, valid AIOS inferences.

- We performed two interesting case studies to justify the merit of the proposed approach: one is to model and infer the AIOS of credit card frauds [3]; the other is to model and infer the AIOS of Internet DDoS attackers [4]. The results are very encouraging.

2.5 How Can ETMA Model IA Metrics and Measurements?

Within the ETMA framework, IA metrics and their measurements are elegantly modeled as the utility earned by either the system or the attacker. In particular,

- IA metrics are used to define utility functions, since the best way to define the system's or the attacker's objectives is using a specific set of IA metrics that can represent the system's overall assurance. The system's utility function defines the overall assurance of the system. And each metric defines one aspect of the system's assurance. On the other hand, the attacker's utility function defines the attacker's overall attack objectives, and each IA metric defines one aspect of the attacker's objectives. Note that each player's *preference* on an IA metric can be modeled as a weight.

- Measurements of IA metrics (before and after an attack) are used to compute the utilities earned by the attacker and the system. The utilities earned by each player are quantified by the measurements of the set of IA metrics that define the player's utility function. Measuring information assurance is about measuring IA metrics, which is elegantly captured by quantitative utilities.

2.6 How Can ETMA Measure IA?

As we pointed out previously, aggregating even a large set of history measurements of a specific IA metric cannot predict the system's assurance (regarding this IA metric) with confidence. The reason is because attacks are intentional, well-planned, and not random. As a result, the actions taken by the attacker when the next attack arrives can be dramatically different from the actions taken by the attacker during his or her previous attacks.

Therefore, we believe that in order to predict assurance with confidence, we must be able to (a) understand non-random attacks, and (b) predict non-random attacks with accuracy. For this purpose, (a) we need to distinguish attacks and attackers; (b) we need to model AIOS; (c) since the attacker's strategies and the system's defense strategies are inter-dependent, we need to model this inter-dependency.

So far, we have proposed a new cyber security model that can capture strategy inter-dependency; we have distinguished attacks and attackers; we have proposed a formal model for AIOS; we have proposed a method to infer AIOS. Now the issue is that after we have done the three things that we need to do, how can we exploit the corresponding results to generate assurance measurements?

The answer is simple. The utilities earned by the system when both the attacker and the system take equilibrium strategies are exactly the set of quantitative assurance capacity measurements we want to get. Why are utilities good assurance measurements? There are a couple of reasons.

- The cyber security model captures the key elements of the attack-defense relationship, such as attack intent, objectives, and strategies; taxonomy of threats, attacks, and vulnerabilities; IA metrics (used to determine the system's utilities); rationality, incentives, costs, utilities; strategy-interdependency; defense postures as the system's strategies; uncertainties and constraints.

- The system's utility function indicates the system's defense objective, so it is not surprising that the system's utility function can be defined by the set of IA metrics that are critical to the system's assurance. Hence, the system's utilities and the system's assurance measurements have the same nature, since they are based on the same set of IA metrics.

- IOS of non-random attacks are properly modeled by the cyber security model. Every AIOS inference is determined by a Nash equilibrium strategy of a specific game between the attacker and the system. Nash equilibrium attack strategies are the "best" attack strategies that the attacker could take. When the system always takes Nash equilibrium strategies, the utilities earned by the attacker when he or she does not take any Nash equilibrium strategy will be smaller than the utilities earned by the attacker when he or she takes a Nash equilibrium strategy. Hence the utilities earned by the system when every player takes a Nash equilibrium strategy represent the *minimum* amount of assurance that the system could get. Therefore, such utilities indicate the *lower bound* of the system's assurance.

The above discussion indicates a simple way to produce quantitative assurance measurements, that is, first compute the utilities earned by the system when Nash equilibrium strategies are taken, then map the utility values to assurance capacity measurements of the system.

2.7 How Can ETMA Predict Attacks?

Attack prediction can be broken down into two categories: trend prediction, which concerns when an attack will happen, and action prediction, which concerns the actions taken by an attack when it really happens. Although the ETMA framework cannot do trend prediction, AIOS inferences can be good predictions about attack actions, since AIOS inferences indicate the attacker's **best** attack strategies. Rational attackers should take the best strategies.

A detailed description of a game theoretic approach to predict attacks appears in [3], where a concrete Bayesian repeated game model is built to predict credit card frauds, and a set of interesting predictions about fraudulent credit card transactions are generated based on a simplified assumption about credit card frauds.

2.8 A Taxonomy of Game Theoretic ETMA Modeling

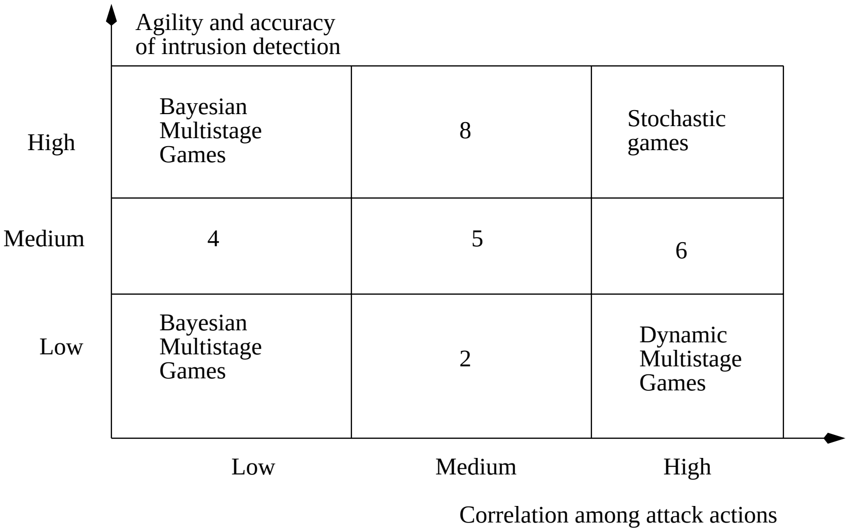
To build a realistic game theoretic assurance measuring model that can produce accurate assurance measurements, the game theoretic measuring model must have the following properties: (a) it must enable the computation of the system's utilities, which indicate the system's assurance; (b) it must make realistic assumptions about the attacker-system relation in order to ensure that the assurance measurements produced are accurate; (c) it must have a realistic knowledge model, since the accuracy of AIOS inferences is heavily dependent on the knowledge model.

A fundamental issue in applying game theory to measure assurance is: "Which type of game theoretic models should be used to measure assurance in a specific attack-defense scenario?" Many types of game theoretic models could be useful in measuring assurance, such as static games, dynamic games, Bayesian games, stochastic games, games with complete information, games with incomplete information, games with perfect information, and games with imperfect information. We found that (a) if the game model is not properly chosen, wrong or misleading assurance measurements can be generated, and that (b) two factors play a critical role in finding optimal assurance measuring game models: one is the agility and accuracy of intrusion detection; the other is the correlation among attack actions. Based on these two factors, we have developed a preliminary taxonomy for game theoretic assurance measuring models, which is shown in Figure 3. Note that for the 'gray' areas, namely regions 2, 4, 5, 6, and 8, usually a tradeoff between the extreme cases needs to be done when we need to build a game theoretic assurance measuring model for such a region. The tradeoffs are dependent on many factors, such as the amount of uncertainty, accuracy, and sensitivity. Readers can refer to [2] for more details and insights about this taxonomy.

2.9 The Impact of ETMA on Security Evaluation

Assurance capacity measurements provide a much more tangible way to evaluate security and survivability; compare the capabilities of two security measures; justify the benefits of novel security measures; and deploy security measures based on the customer's requirements.

Within the ETMA framework, the security of a system can be evaluated based on either individual IA metrics or a weighted combination of a set of IA metrics. The ETMA framework can generate the assurance capacity measurements for every IA metric that is desired to eval-



Agility and accuracy of intrusion detection	Low	Medium	High
High	Bayesian Multistage Games	8	Stochastic games
Medium	4	5	6
Low	Bayesian Multistage Games	2	Dynamic Multistage Games

Figure 3: A Taxonomy of Game Theoretic Assurance Measuring Models

uate the security of a system. Each such IA metric indicates one aspect of the security of the system, and a weighted combination of a set of IA metrics can represent the overall security of the system. These quantitative measurements quantify the security of a system in a much more tangible way, especially from a comparative stand point of view. Qualitative evaluation sometimes cannot tell the difference between two systems of the same security level, but quantitative evaluation can.

Within the ETMA framework, security evaluation can be performed either during design time or during run time. During design time, we can get assurance measurements about a specific security design based on an assumption about the environment and the attackers. After a secure system is deployed, during run time, we may need to adjust the assurance measurements we get during the design time since the assumptions we made before may not be very consistent with the real world situations. The intelligence gathered and the experiences gained during run time enable us to refine our knowledge base, and the "assumed" attacker's knowledge base. And as a result, such adjustment can produce more accurate assurance measurements. Moreover, the refined knowledge may enable the system to do better defense.

2.10 The Impact of ETMA on Secure Systems Design

So far, we considered the following question: “Given a system protected by a set of security measures, what are the AIOS to attack the system and how assured is the set of security measures?” In this section, we are going to address the *dual* question: “Given the ability to measure assurance, which set of security measures can achieve the largest amount of assurance?”

Traditional security design methods are ad hoc. They are based on people’s experiences and what feels right. The designer is usually not clear about the amount of assurance that could be achieved by his or her design; thus the designer cannot guarantee that the customer’s assurance requirements will be satisfied.

The ETMA framework can result in a systematic methodology in optimal security design since (a) the ability to quantitatively measure the assurance of a security design enables the goal of every security design effort to be precisely defined, whether the goal is “achieving the largest amount of assurance” or “satisfying a set of specific assurance requirements of the customer”. (b) The ability to measure assurance indicates the ability to compare two security designs and tell which one is better. (c) The ability to compare two security designs indicates a simple generate-and-test approach to optimal security design, which is as follows. Of course, this approach can be improved in a variety of ways. For example, expert knowledge and a set of validated design principles can make the generate-and-test process converge in a much quicker way.

1. Propose an arbitrary design.

2. Measure the assurance of this design.

3. Change one factor of the design and repeat steps 1 and 2.

4. Compare the new design and the original design, and pick the better one.

5. Repeat steps 3 and 4 many times until the design satisfies the customer’s requirements or the design cannot be further improved.

2.11 Case Study 1

In [3], we have done a case study to measure the assurance capacity of credit card authorization systems armed with a fraud-detection sub-system. We developed a specific Bayesian game formalization to model the battles between frauds and a credit card company. This formalization develops a novel probabilistic uncertainty model between the fraud and the credit card company. Based on a simplified assumption about credit card frauds, the case study yields interesting measurements about the *maximum amount of money* that a customer could lose.

2.12 Case Study 2

In [4], we have finished Phase I of an assurance capacity measuring case study regarding Internet DDoS attacks. The goal of this case study is to measure the assurance capacity of pushback-based Internet defense in terms of the *minimum bandwidth* of good traffic to the target. We use *ns2* to simulate both UDP and ICMP based DDoS attacks on a set of routers armed with the pushback module. We developed a specific Bayesian game formalization to model the battles between the DDoS attacker and the set of routers. Using the (simulation) measurements measured based on a set of settings of router and (bad/poor/good) traffic, Phase I yields interesting assurance capacity measures when the defense strategy is rather fixed. In Phase II, we plan to incorporate a large space of defense strategies.

2.13 The Main Technical Issues of ETMA

In order to successfully develop the ETMA framework, we need to address a variety of intriguing technical issues, which can be organized through five dimensions.

- The modeling dimension: taxonomy of game theoretic models; IA metrics; utility functions; roles of threat, vulnerability, and attack taxonomies; knowledge and belief; dynamic knowledge; knowledge inference; uncertainty; attack statistics; component level modeling; composite modeling; effect-based modeling.

- The inference dimension: Nash equilibrium strategies; computational complexity; accuracy; real time inferring; sensitivity; approximate inference methods.

- The systems dimension: OS; database systems; middleware systems; distributed systems; computer networks; wireless; programming systems; networked information systems such as JBL and network-centric warfare. How can ETMA handle large scale networked information systems? How to demonstrate the set of assurance measuring technologies in the context of an actual problem?

- The life cycle dimension: design time; run time.

- The attack dimension: component attacks (e.g., DDoS, malicious transactions); composite attacks; novel attacks.

2.14 Technology Development Roadmap

The ETMA framework technology development roadmap is shown in Figure 4. In particular,

12 months: (1) The taxonomy of game theoretic formalizations for AIOS modeling and assurance capacity measuring developed. (2) A theoretical, incentive-based, economic model of AIOS developed. (3) A general game theoretic formalization for assurance capacity measuring developed.

18 months: (1) Case study 1 (regarding credit card frauds) finished with a specific game theoretic formalization and a set of measurements about the fraud's attack capacity, the credit card company's defense strategy, and the assurance capacity. (2) Case study 2 (regarding Internet DDoS attacks) finished.

24 months: (1) The impact of incomplete information and uncertainty (IIU) on the accuracy of assurance capacity inference investigated, modeled, analyzed, and measured through probabilistic IIU models, Bayesian and stochastic game theory, and modal logic based uncertainty reasoning. (2) IIU models integrated into the game theoretic formalization for assurance capacity measuring. (3) Case study 3 (regarding self-healing wireless ad hoc routing) finished.

36 months: (1) A practical (real-time) assurance capacity inference machine developed through efficient approximate, incremental algorithms for (game theoretic) economic computing whose complexity is typically NP hard. (2) The approximate algorithms exploit attack

	0 - 12 months	12 - 24 months	24 - 36 months	36 - 84 months
The modeling dimension	Taxonomy of game models AIOS modeling Game theoretic models for assurance measuring	Component level modeling IA metrics; Utilities Incomplete info and uncertainty Dynamic Knowledge Knowledge inference Attack statistics	Effect based modeling	Large scale composite modeling
The inference Dimension	Nash equilibrium strategies	Accurancy Complexity Sensitivity	Real time inferring Approximate methods	Large scale approximate inferring
The systems dimension & dimenstration		Case study 1 (middleware, DB) Case study 2 (Internet) Demo	Case study 3 (wireless ad hoc routing) Demo	OS Distributed systems Programming systems Large scale networked info systems such as JBI Demo
The attacks dimension	Component attacks		Novel attacks Composite attacks	Composite attacks on complext applications
The life cycle dimension	Design time assurance measuring	Run time assurance measuring	Optimal security design Quantitative security evaluation	

Figure 4: ETMA Roadmap

and defense semantics to substantially reduce the computational complexity without losing a lot of inference accuracy. (3) The trade-offs between computational complexity and inference accuracy investigated quantitatively. (4) Case study 4 (regarding effect-based attacks) finished. (5) A methodology for optimal security design developed. (6) A methodology for quantitative security evaluation developed.

3 A Theory of QoIA Management

3.1 Overview

Attack resilient systems extend traditional secure systems to *survive* or *operate through* attacks. The focus of survivable and self-regenerative systems is the ability to continue delivering essential services in the face of attacks. New mechanisms of attack resilient systems include but not limited to intrusion detection, fragmentation, replication, migration, masking, isolation, containment, recovery, and self-regeneration.

The area that we find especially interesting is *QoIA* (Quality of Information Assurance) management. The ability to continue delivering essential services to applications (and users) in the face of attacks suggests that the delivered essential services must be *valid* or not (seriously) distorted by the attacks, since making the system continuously deliver distorted (or invalid) services can be the goal of the attacker. From the perspective of trusted computing, the *validity* of a delivered service can be measured by the QoIA associated with the service. The QoIA associated with a service indicates the extent to which we *trust* that the service is not distorted or corrupted by the attacker. Services associated with higher QoIA are more valid than those associated with lower QoIA. We call services associated with a specific level of QoIA *QoIA services*.

The concept of QoIA services is closely related to the concept of *trustworthiness* (or assurance). The concept of trustworthiness has two important aspects: one is the extent to which we trust that the system *state* is valid and not corrupted (by attacks). We call this aspect *state trustworthiness*. The other is the extent to which we trust that the services delivered by the system are not corrupted, namely, QoIA services. We call this aspect *service trustworthiness*. State trustworthiness represents the system security officer's view of the system's trustworthiness, while service trustworthiness represents the services' or the users' view of the system's trustworthiness.

In general, a system with a higher level of state trustworthiness should be able to yield better service trustworthiness. However, although at one point of time the system has a unique level of state trustworthiness, different services or users can have very different *views* of the system's trustworthiness. For example, consider a bank database application where two users, Alice and

Bob, are executing two *withdraw* transactions (i.e., services), denoted T_A and T_B respectively. If Alice's account is corrupted but Bob's is not, then Alice's (or T_A's) view of the system's trustworthiness can be "very bad" while Bob's (or T_B's) view of the system's trustworthiness can be "very good".

The primary goal of attack resilient systems is to deliver required QoIA services in the face of attacks. However, existing attack resilient systems have five major drawbacks in delivering QoIA services. First, they focus on state trustworthiness and cannot *differentiate* the trustworthiness requirements from different services on different data objects. (For example, in a bank the trustworthiness requirements of services on a customer's home address can be much less restrictive than those on the customer's account balance). This drawback can place the system in a dilemma between satisfying every customer and saving cost. Providing (on average) the same level of trustworthiness to services with different QoIA requirements can *waste* a lot of resources.

Second, (most) existing attack resilient systems deliver QoIA services in a vague, qualitative way. As a result, neither the applications (users) nor the system can tell the other party the *exact* level of (service) trustworthiness they ask for or can provide. This drawback disables the system to deliver *quantitative* QoIA services.

Third, due to the changing *environment* of the system, the system's defense *behavior* must *adapt* to the environment changes in order to keep the ability to continuously deliver the required QoIA services. This process is called *QoIA adaptations* (or reconfiguration). QoIA adaptations are necessary because if the system is not adaptive, quickly changed attack patterns and workloads could seriously jeopardize the state trustworthiness and correspondingly the service trustworthiness. However, based on a qualitative understanding of trustworthiness, existing attack resilient systems cannot do *quantitative* QoIA adaptations.

Fourth, (most) existing attack resilient systems cannot deliver QoIA services in a cost-effective way. Maintaining (service) trustworthiness consumes resources. Cost-effective QoIA adaptations must be able to trade off trustworthiness versus cost. Adhoc tradeoff mechanisms, however, could yield poor cost-effectiveness. Quantitative tradeoff mechanisms are needed.

Fifth, existing QoIA adaptations are *passive*. That is, every adaptation activity is triggered by some effects of attacks, and the system always *lags behind* the attacker. As a result, although existing QoIA adaptations are effective in the middle of a wave of attacks, they can fail when the wave of attacks (suddenly) ends (or starts), since their agility is not good enough to handle suddenly changed environments. *Active* QoIA adaptations are needed.

In this seedling project, we advocate a quantitative, active, differential QoIA adaptation approach to attack resilient systems. The objective of this effort is to build a new paradigm for attack resilient systems that is able to continue delivering required QoIA services in the face of attacks. A potentially feasible solution is to deliver differential, quantitative QoIA services through *predictive* QoIA adaptations where *predictions* of the environment changes are exploited to enable active defense, and the trade-off between trustworthiness and cost is done in a smooth way.

As we pointed out previously, the ETMA framework builds the foundation for our QoIA theory. Without the ability to measure assurance, quantitative QoIA services cannot be delivered.

3.2 How Can the Theory Exploit IA Measurements?

Two types of assurance measurements are included in the QoIA theory:

- State assurance measurements, which measure the assurance of a system from the system security officer's point of view. This view focuses on the state of each component of the system. A component can be a software component or a hardware component. A component can be a piece of executable code or a piece of data processed by a program. One system component, no matter how many services or users are using it, has a single state, which indicates how 'healthy' the component is.

- Service assurance measurements, which measure the assurance of a system from the services' or the users' point of view. This view focuses on the amount of assurance associated with a service delivered by the system. Service assurance concerns the extent to which a service can be distorted.

It should be noticed that state assurance and service assurance are closely related to each other. The assurance of a service is dependent on the state assurance of each component involved in the service. For example, in a database application, the extent to which a transactional service is distorted is dependent on the extent to which the data items read by the service are corrupted.

To deliver QoIA services, we need to be able to do both *static measuring*, which focuses on one point of time, and *dynamic measuring*, which concerns a period of time. Note that dynamic measuring tells much more about survivability.

3.3 The Problem Space

The problem space can be simply modeled as a QoIA service provision process:

- QoIA service reservation. How to specify security and survivability requirements? How to quantitatively specify such requirements?

- Mapping service assurance requirements to state assurance requirements. Direct delivery of QoIA services based on service assurance measurements can be too expensive to be practical. A realistic approach can have two steps: (1) map the set of service assurance requirements to a set of state assurance requirements; (2) deliver the set of QoIA services indirectly by maintaining the state assurance in such a way that the set of state assurance requirements will not be violated.

- QoIA service provision. After the set of service assurance requirements are mapped to a set of state assurance requirements, how to maintain the state assurance in such a way that the set of state assurance requirements will not be violated? The basic approach is state-assurance-based QoIA adaptations. How to do QoIA adaptations in such a way that the state assurance requirements will be satisfied? How to do intelligent QoIA adaptations? How to do composite QoIA adaptations? How to do differentiated QoIA adaptations? How to do predictive QoIA adaptations? How to handle large-scale networked information systems?

- QoIA service validation. How to validate the effectiveness of a QoIA aware attack resilient computer system in delivering QoIA services? How to measure a system's capacity in delivering QoIA services?

3.4 How Can the Theory Specify Security and Survivability Requirements?

We outline a *white box* approach. We model a service as a white box with a set of *inputs*, denoted $\{i_1, i_2, ..., i_k, ...\}$, and a set of *outputs*, denoted $\{o_1, o_2, ..., o_k, ...\}$. We say the box is *white* because we assume the source code of the service is known. The QoIA associated with the service can be measured by the *validity* of the outputs. So the QoIA requirements can be specified as $\{v_{o_1}, v_{o_2}, ..., v_{o_k}, ...\}$ where v_{o_i} is the required validity level on o_i and o_i indicates a data element. In our model, v_{o_i} is the required (lower-bound) *probability* that o_i is valid (i.e., not damaged), that is, when the service is executed, the probability that o_i is valid should not be less than v_{o_i}. v_{o_i} represents the user's view of the *importance* of o_i. Note that although v_{o_i} can be more accurately measured by how o_i is different from the value of this output when there are no attacks, value-based v_{o_i} requirements are very complicated and difficult to satisfy.

For a service, the mapping from its QoIA requirements to its state trustworthiness requirements is the mapping from $\{v_{o_1}, ..., v_{o_k}, ...\}$ to $\{v_{i_1}, v_{i_2}, ..., v_{i_k}, ...\}$. This mapping can be done based on the *control flows* inside the service, and the amount of influence of i_p on o_q (which indicates the extent to which o_q is affected by i_p). For example, if a service does the following: $o_k = i_1 + i_2$, and v_{o_k} is p, then if i_1's and i_2's influences on o_k are the same, it can be a good idea to let both v_{i_1} and v_{i_1} be $p/2$. To develop the mapping algorithm, the control flows and the *semantics* of service operations can be exploited. Although analyzing service programs requires some effort, for every service type we need to develop only one mapping algorithm.

After the QoIA requirements of every service are mapped to a set of state assurance requirements, the assurance requirements on a data element x can be specified as $\{v_x^1, v_x^2, ..., v_x^n\}$ where v_x^i is the assurance requirement of service i on x. Since we need to satisfy all of them, the combined $v(x)$ is the *maximum* probability within this set.

Till now, we have transformed the set of differential QoIA requirements to an equivalent set of element level state assurance requirements. Next, we can deliver QoIA services through dynamic, adaptive state assurance maintenance.

Example. In a QoIA aware attack resilient database system, we can specify QoIA requirements from the data integrity perspective as follows. A database is a set of data objects. It is denoted as $DB = \{o_1, o_2, ..., o_n\}$. Each database provides a set of services to its users. A service S_i is a set of database transactions, denoted as $S_i = \{T_1, T_2, ..., T_k\}$. A user is a business customer who accesses the database and possibly changes the database state through services.

For a service that user u_i asks for, u_i's QoIA requirements on the service are specified thorough the set of transactions included in the service. In particular, u_i's QoIA requirements on a transaction T_j are specified as follows. For each data object o_i read by T_j, a lower bound of the integrity level of o_i is specified and denoted as $th(o_i^r, T_j)$, which means that in order to satisfy the QoIA requirements of the service, the integrity level of o_i cannot be lower than $th(o_i^r, T_j)$ when T_j reads o_i.

The integrity level of a data object o_i indicates whether o_i is corrupted or not. When a database is attacked by a malicious transaction, although there is an Oracle who knows exactly if o_i is corrupted or not at any time, the database system does not have this knowledge until a detailed damage assessment is done. Since many large-scale database systems critical to businesses are expected to be available continuously and can only be stopped for repair at great cost, we do not want to stop delivering services until the damage assessment is finished. For this purpose, we adjust our definition of integrity level as follows. In our model, at one point of time, the integrity level of o_i is defined as the *probability* that o_i is not corrupted. We call $p(o_i)$ the *integrity level* of o_i where $0 \leq p(o_i) \leq 1$. Note that $p(o_i)$ can be estimated based on the transaction history in a variety of ways.

As a result, a user's QoIA requirements on a service S_i is specified as a set of probabilistic integrity level thresholds associated with the set of data objects that the service will access. The database system satisfies the user's QoIA requirements on S_i at time t_i if the integrity level of every data object o_i read by a transaction T_j of S_i is not below $th(o_i^r, T_j)$.

3.5 How Can the Theory Deliver QoIA Services?

A tentative framework of QoIA management is shown in Figure 5. In general, the changing environment is monitored, and the environment changes will trigger the system to adjust its defense *posture* under some conditions. The objective is to make the adjusted defense posture more effective in delivering QoIA services. In particular, the *QoIA Reservation Console* is the interface for users to *reserve* QoIA services (i.e., to specify their QoIA requirements). The *Observer* collects useful known measurements about the *environment*. The *Trustworthiness*

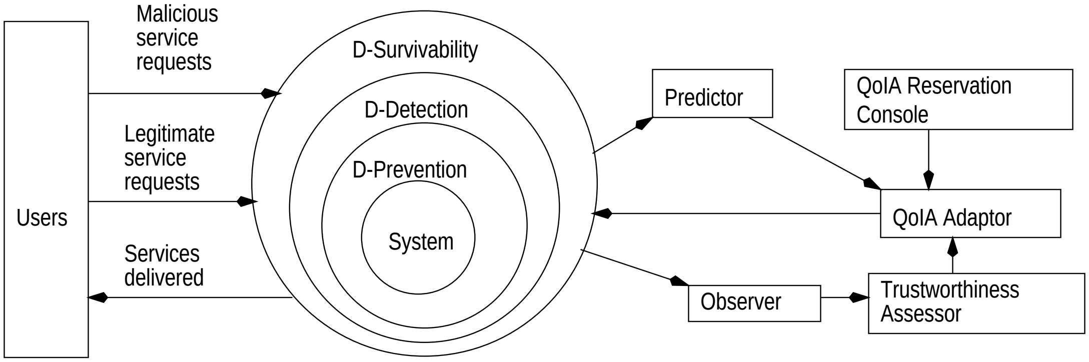


Figure 5: QoIA Management Framework

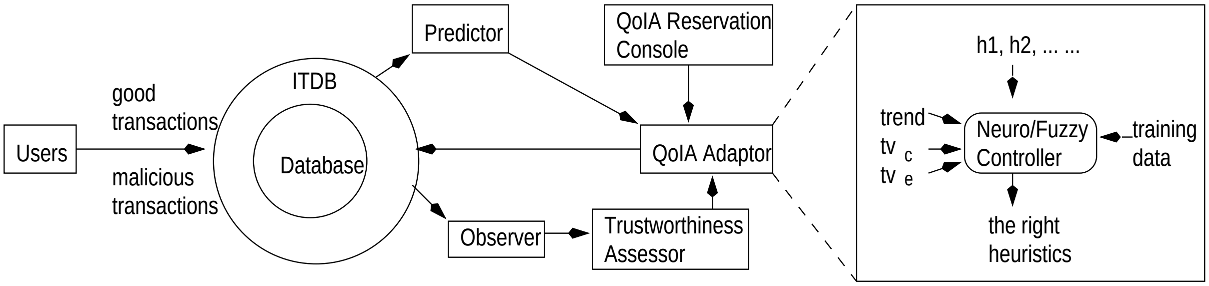


Figure 6: QoIA Aware Database Systems Architecture

Assessor assesses the state trustworthiness of the system. The *QoIA Adaptor* does (intelligent) QoIA adaptations. Finally, the *Predictor* is used to do predictive QoIA management. As we mentioned previously, a key technology in QoIA management is differentiating security and survivability controls. This is why our prevention measures are differentiated; our detection measures are differentiated; and our survivability measures are also differentiated. Finally, it should be noticed that this generic framework is system independent.

Based on the generic QoIA management framework, specific QoIA management schemes for specific information systems can be developed. For example, a QoIA management scheme for attack resilient database systems is shown in Figure 6. Note that ITDB is a set of database intrusion tolerant measures we have developed recently [1].

In this scheme, QoIA adaptations are triggered in three situations: (1) when the Observer raises alarms about the database's health level; (2) the SSO can directly activate some adaptation operations; (3) otherwise, the scheme breaks the time into a sequence of *adaptation intervals*, denoted AI_1, AI_2, ..., AI_k, and does adaptations periodically at the end of each adaptation interval.

In this scheme, at time t, the *environment* is determined by tv (the state trustworthiness vector)[*], the *workload* (on the database) which produces the transaction *history*, and the attacks. When there are a lot of attacks, we say the environment is *hostile*.

In this scheme, almost every ITDB component is reconfigurable and the *behavior* of each ITDB component is *controlled* by a set of *parameters*. For example, the major control parameters for the Intrusion Detector are TH_m, which is the threshold used to report malicious transactions, and TH_s, which is the threshold to report suspicious transactions. The major control parameters for the *Damage Container* is the amount of allowed damage *leakage*, denoted DL. When $DL = 0$, multi-phase containment is enforced; when there is no restriction on DL, one-phase containment is enforced. The major control parameters for the *Policy Enforcement Manager* is the transaction delay time, denoted DT. When $DT = 0$, transactions are executed in full speed; when DT is not zero, transaction executions are slowed down. At time t, we call the set of control parameters (and the associated values) for an ITDB component C_i the *configuration* (vector) of C_i at time t, denoted $cv_t(C_i)$. Assume $ITDB = \{C_1, C_2, ..., C_n\}$, then $[cv_t(C_1), ..., cv_t(C_n)]$ is called the *configuration* of ITDB at time t. QoIA adaptations are done by adjusting ITDB from one configuration to another configuration. QoIA adaptations should be done in such a way that the set of QoIA requirements will not be violated. Since it is not difficult to see that the optimal adaptation problem is a NP hard problem, we propose a neuro-fuzzy controller to learn and enforce the best heuristic QoIA adaptation operations.

3.6 The Main Technical Issues of QoIA Management

- State and service assurance modeling and assessment.
- The concept of QoIA services. The architectures of QoIA management.
- Maintaining service assurance through state assurance maintenance.

[*] We use a vector of IA metrics, denoted $[m_1, m_2, ..., m_k]$, to measure the state trustworthiness of the database system.

- QoIA adaptations.
- Differentiated QoIA management and service assurance maintenance. Existing security and survivability mechanisms are *uniformly* enforced on all system components. For example, when we switch ITDB from one-phase damage containment to multi-phase containment (via an adaptation), every table in the database will be contained based on timestamps. However, this *uniformity* can significantly reduce the cost-effectiveness of QoIA management due to the fact that differential QoIA services usually cause different state trustworthiness requirements on different data items. To illustrate, consider the scenario when the state assurance requirements on table 1 is much more restrictive than those on table 2. To satisfy the assurance requirements on table 1, we may need to switch ITDB to multi-phase containment. However, although multi-phase containment is necessary to maintain the required integrity level of table 1, one-phase containment can be good enough to maintain the required integrity level of table 2, and enforcing multi-phase containment also on table 2 can unnecessarily cause extra availability losses.

  The above example suggests that enforcing *uniform* survivability control on all data items can cause extra cost, availability losses, or integrity losses without enhancing our ability to deliver QoIA services.

  To further improve cost-effectiveness, we suggest a differentiated approach where different security and survivability controls are enforced on different system components according to the different state assurance requirements on these system components. Note that differentiated QoIA management can be enforced at multiple granularities.

- Predictive QoIA management. Most existing dynamic configuration mechanisms are *passive*. That is, every adaptation activity is triggered by the effects of attacks, and the system always lags behind the attacker. As a result, although passive QoIA adaptation mechanisms are effective when the environment changes with a consistent *trend* (e.g., in the middle of a wave of attacks), *sudden* environment changes (e.g., when the wave of attacks ends or starts) can seriously jeopardize the effectiveness of passive QoIA adaptations due to the adaptation latency. During the adaptation latency, if the old environment and the new environment are similar, then the current configuration of the system can work well. However, if these two environments are very different, the current configuration of the system can perform very poorly before the adaptation is done. That is, sudden environment changes can cause serious assurance losses during the adaptation latency.

We suggest an *active* defense approach to address the above problem. In particular, we keep on monitoring and predicting the *trend* of both the attacks and the workload. Then the *predictions* are exploited to achieve a much *smoother* adaptation in face of sudden environment changes. Predictions can be very helpful because the adaptation latency can be significantly reduced.

- How to handle large scale networked information systems? How to deliver composite QoIA services? How to make QoIA management scale up?
- QoIA aware database systems; QoIA aware distributed and P2P systems ; QoIA file and Internet services; QoIA aware networking; QoIA aware mobile computing; QoIA aware systems of systems; QoIA aware large-scale networked information systems such as Joint Battlespace Infosphere.

3.7 Technology Development Roadmap

The QoIA management technology development roadmap is shown in Figure 7.

4 Conclusion

This seedling project has two objectives: one is to explore an economics theoretic framework for measuring assurance; the other is to explore a theory of QoIA management. In this paper, we summarize the results we obtained from the seedling effort. Regarding each objective, we focus on (a) defining the problem space; (b) offering some potentially feasible solutions; (c) creating a technology development roadmap for a 5 to 7 year time horizon. In addition, we have performed several case studies, and the results are very encouraging.

	0 - 12 months	12 - 24 months	24 - 36 months	36 - 84 months
State & service assurance modeling and measuring	A Method for service assurance measuring Statistics-based state assurance measuring	Mapping server assurance to state assurance	Measuring a system's capacity in delivering QoIA services	Composite service assurance measuring
QoIA service reservation	QoIA requirement specification in DB & networking	QoIA service reservation console		Taxonomy of QoIA services
QoIA service provision & demonstration	QoIA Management framework Intelligent QoIA adapatations Differentiated QoIA adapatations	QoIA Aware DB & demo Compsite QoIA adaptations	Predictive QoIA adaptations QoIA aware mobile ad hoc routing & demo ETMA based QoIA aware system design	Large scale composite QoIA management QoIA aware large scale NIS such as JBI
QoIA service validation		Evaluation of QoIA Aware DB	Evaluation of QoIA aware MONET	Validation of QoIA aware complext applications

Figure 7: QoIA Management Roadmap

References

[1] P. Liu. Architectures for intrusion tolerant database systems. In *Proc. 17th Annual Computer Security Applications Conference*, 2002.

[2] P. Liu. Incentive-based modeling of attacker intent, objectives, and strategies in cyber security. Technical report, School of IST, Penn State University, 2003.

[3] P. Liu and L. Li. A game theorectic approach to attack prediction. Technical report, School of IST, Penn State University, 2002.

[4] W. Zang and P. Liu. Measuring the internet's resilience against ddos attacks. Technical report, School of IST, Penn State University, 2003.